

**Kyrgyz-German Institute of Applied Informatics**

**Course Work**

**«Education Center»**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subject’s Information:** | | | | |
| Subject | Programming languages | | | |
| Session | May 2021 | | | |
| Lecturer | Askarbek Almazbek uulu  Karagulova M.  Asanova A. | | | |
| Coursework Type | Team Project | | | |
| Hang-out Date | 06.02.2021 | | | |
| Due Date | 20.04.2021 | | | |
| **Students’ information** | | | | |
| **Name** | | **ID** | **Contract** | **Sign** |
| Beishenova Saltanat | | AIN-1-20-142-1 | 996557909340  saltanatb02@gmail.com |  |
| Azamatov Eldar | | AIN-1-20-136 | +996 778 530 751  azamatov3333@gmail.com |  |
| Ustemirov Mars | | AIN-3-20-187-4 | +996706295616  mars99879@gmail.com |  |
| Rustamov Artur | | AIN-1-20-153 | 0500811882  Rustamovartur675@gmail.com |  |

**Table of Content**

|  |  |
| --- | --- |
| Content | Page number |
| Team Roles | 3 |
| Architectural Goals and Constraints | 4 |
| Gannts Chart | 5 |
| Software requirements specification | 6 |
| Graphic representation | 7 |
| Use Case diagram | 11 |
| Database Review | 13 |
| Functionality | 15 |
| Vision | 26 |
| Risk Management | 27 |
| Conclusion. Code Review | 28 |

**Team Roles**

Due to final exams of second semester in Kyrgyz-German Institute of Applied Informatics (INAI.kg) we were tasked to make a coursework for Programming Languages subject. Our theme is “System of educational center”.

Azamatov Eldar:

* Design
* Functional part of the code

Beishenova Saltanat:

* Functional part of the code
* Presentation

Ustemirov Mars:

* Functional part of the code
* Documentation

Rustamov Artur:

* Data Base Architecture
* Presentation

**Introduction**

**Architectural Goals and Constraints**

There are several goals which we were led by. At the same time creating the code we faced some problems and constraints that did not give us the opportunity to write more complicated work.

Goals:

By users of the program, teachers of the educational center, students,

director. Everyone will have their own access to the system by keyword.

Depending on the category of the account of a person, opens

the corresponding menu:

* General information about the number of students, student attendance, the schedule of classes, the number of subjects with their names is visible and accessible only to the principal and teachers in the relevant subjects.
* The teacher has the ability to see the number of students, their attendance and evaluate or change grades appropriately for a particular student.
* The student has the opportunity to see through his account his visit, his grade for each item.

Constraints:

We can face up with several problems due to our lack of experience and knowledge.

1. We are adding interface to our software, that is why we can have some bugs or the interface can be not fully completed.
2. We are using new data base system that is why some problems with SQL Calls can appear.
3. This is a group course work, so we have had problems with uniting the code and documentation.

Gantts Chart

A screenshot of a computer

Description automatically generated with medium confidenceGraphical user interface, table

Description automatically generated

The Diagram above shows the Gantt Chart of the time taken for the team and it’s process on creating a program for Educational Center System, Inside consists of Meetings, Time taken for program to work, Design, and the Final preparation of work such as doing report, doing Vision, Risk analyses and etc.

Software requirements specification

*Overall Description \ Product features*

“Educational System” is a software that will give different level of access to the database for three accounts to give the opportunity to manage the system or get the information needed.

Regulatory and reference information of the automated information system "Educational System" is presented in the form of a reference book for all types of accounts in the form of one of the menu items, by clicking on which long text is displayed as instructions.

*Overall Description \ Design and implementation constraints*

*Programming Language*: Java

*IDE:* Intellig IDEA by JetBrains

*Database*: mySQL

*Interface*: JavaFX

User Description:

You should install mySQL to have access to the database.

*Overall Description \ Operating environment*

This program is compatible with JDK 7+ versions. Any JDK older than this might have some issues with compatibility, but the program can be reverting to older version.

Graphic Representation of the architecture:

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

In this diagram there is shown an architectural representation of the system including detailed structure of each account. There is three main accounts: student, teacher and director. First of all they must authorize themselves. Only person with account of director can add and delete students and teachers account and input their login and password. Students and teachers account can authorize by inputting login and password that were inputted by director when he was adding new account.

Students account have eight main functions:

1. Show the list of subjects - shows all subjects in certain educational center
2. Show the list of exercise - shows all exercises given by teacher of certain subject
3. Show the list of exams - shows start and end date of exam
4. Show the list of offsets - shows the list of offsets
5. Show the max score - shows max score by certain subject
6. Show the min score - shows min score by certain subject
7. Show the list of scores - shows all scores for all subjects
8. Exit program -returns program to authorization page

Teacher account have eight main functions:

1. Show the list of subjects - shows all the subjects taught by the teacher
2. Show the list of students - shows all the students he teaches and their grades
3. Show the list of exams - shows start and end date of exam
4. Show the list of offsets - shows the list of offsets
5. Show the students max score - shows max score of all students teacher taught
6. Show the students min score - shows min score of all students teacher taught
7. Show the list of scores - shows all scores of all students
8. Exit program -returns program to authorization page

Directors account have eight main functions:

1. Show the list of subjects - shows all subjects in certain educational center
2. Shot the list of teachers - shows all exercises given by teacher of certain subject
3. Add new teacher - adds new teacher
4. Delete teacher - deletes existing teacher
5. Show the list of students - shows all students in certain educational center
6. Add student - adds student
7. Delete student - deletes student
8. Exit program -returns program to authorization page

Use-case diagram:

Diagram

Description automatically generated

In the Use Case, I drew out the Possible outcomes of the interaction of systems between Main Workers. This is a representation of the overall program functionalities when is used between student, teacher and user.

1. Teacher
2. Students
3. Subjects
4. Grades
5. Home tasks
6. Groups
7. Student and subject relationship
8. Student and home tasks relationship

All classes store and save information according to input from the application and queries that will be shown in the code review. We designed our architecture to be easy to use, and more suitable for educational institutions and corresponding to the terms of reference of the course work.

Database architecture:

A picture containing table

Description automatically generated

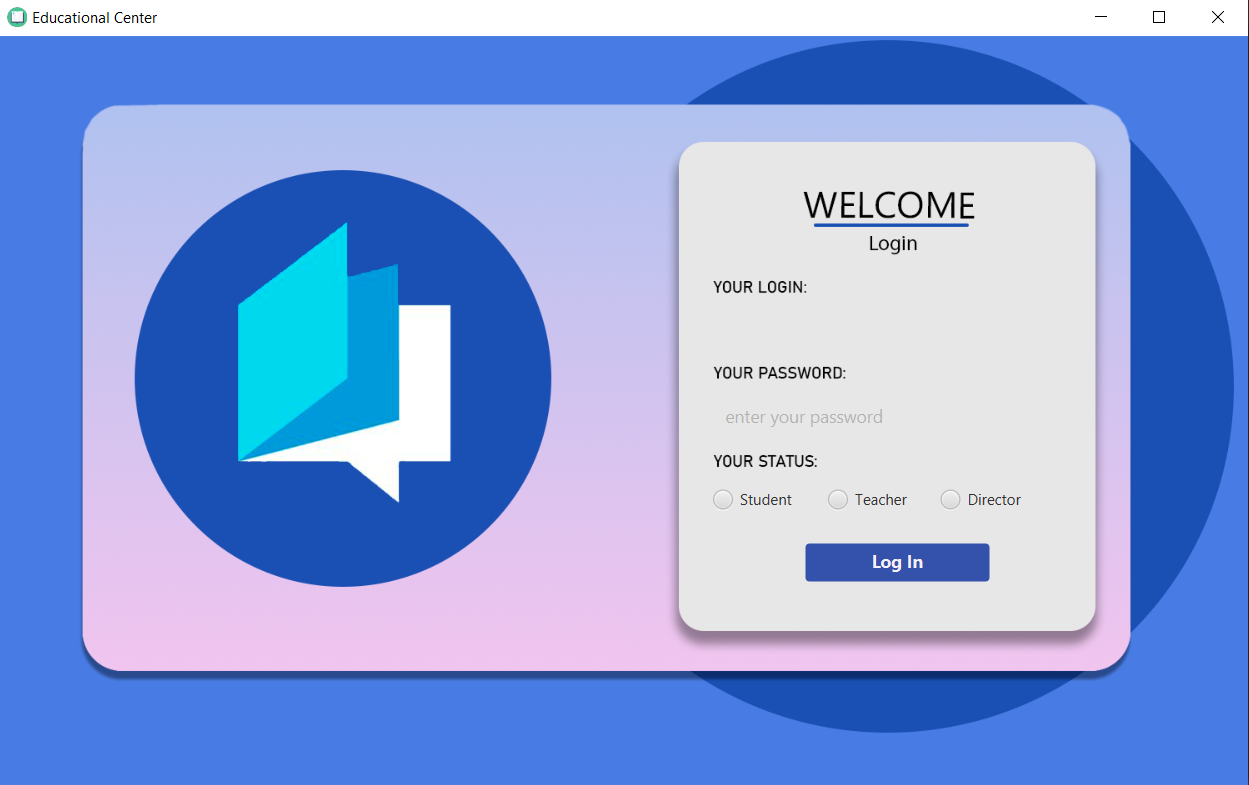
UML Class diagram is drawn out to help us to further understand what is needed in the database and how are they linked to each other. By having this UML class diagram as a guideline we have a clear mind set on how the database should be set up and the implementation of tables to later connect into our system.

1. Teacher
2. Students
3. Subjects
4. Grades
5. Home tasks
6. Groups
7. Student and subject relationship
8. Student and home tasks relationship

All classes store and save information according to input from the application and queries that will be shown in the code review. We designed our architecture to be easy to use, and more suitable for educational institutions and corresponding to the terms of reference of the course work.

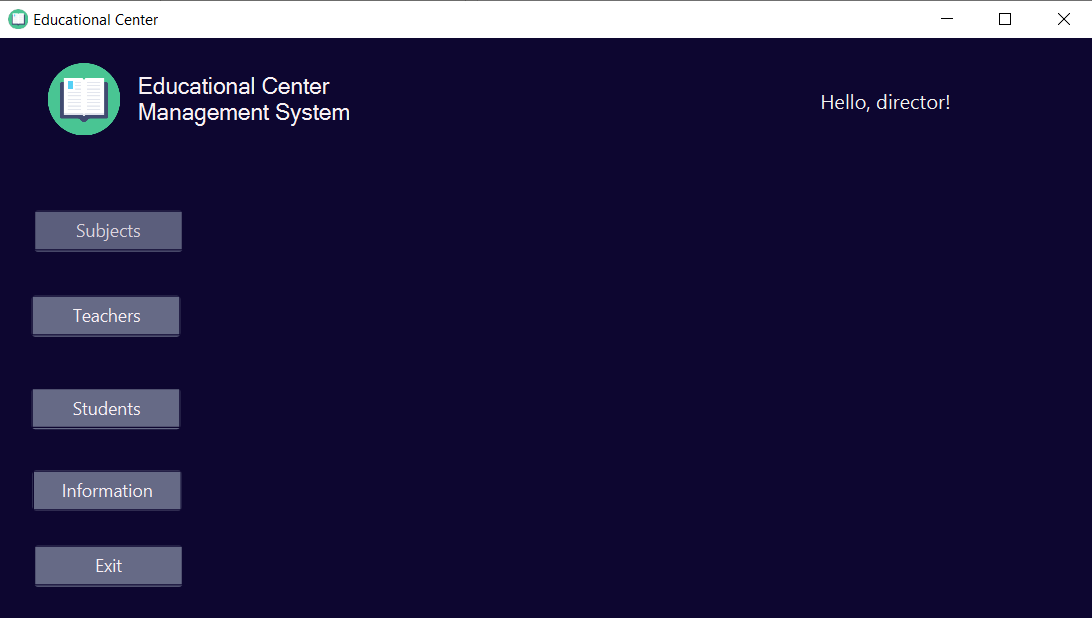
Functionality:

This subsection contains the requirements for the program and outlines the functionality of each step when the user interacts with the system. All actions performed by the user and how it works are described sequentially.



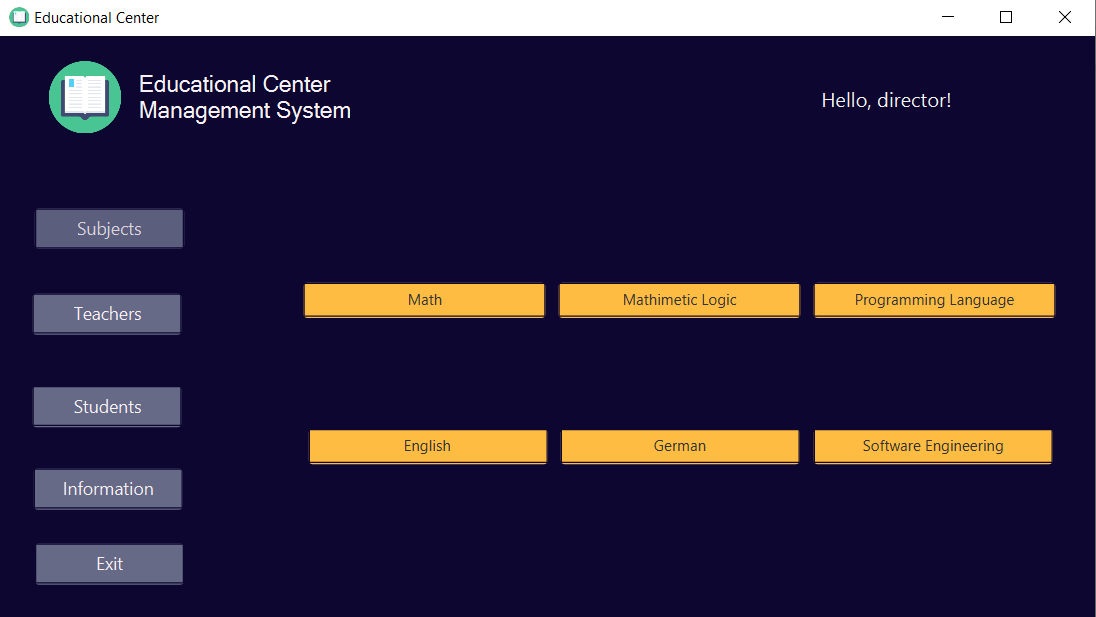
Authorization page

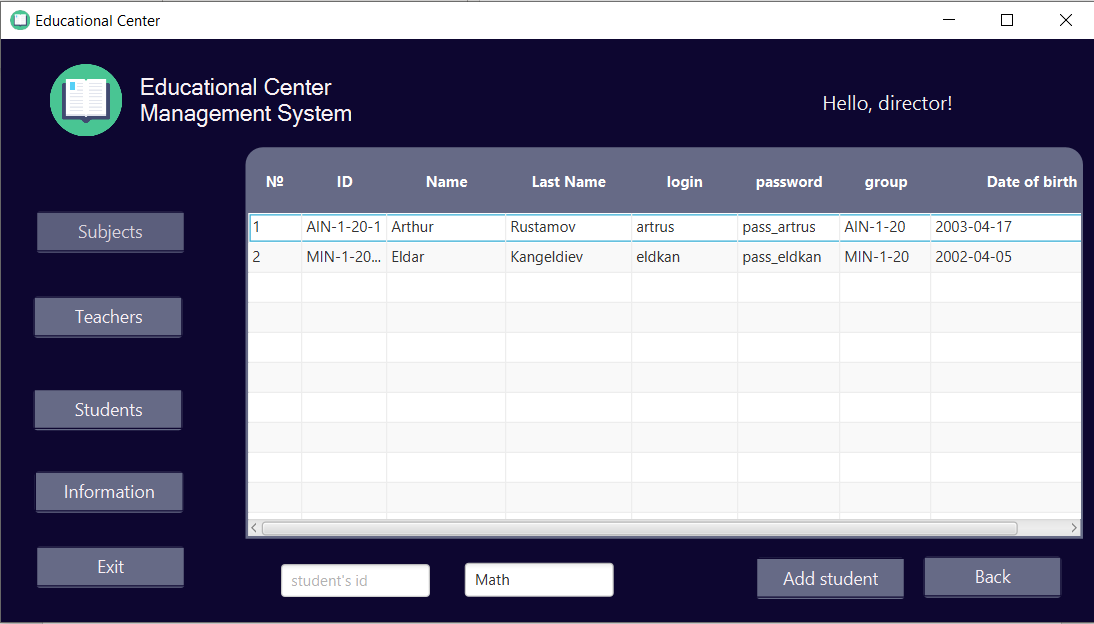
On authorization page you can see login and password inputs, log in button and three points for choosing type of account. User can choose the type of his account and log in if his login and password matches to the keys in the database.



The page of the director account

On this page you can choose the function you want, and it will show you another scene.

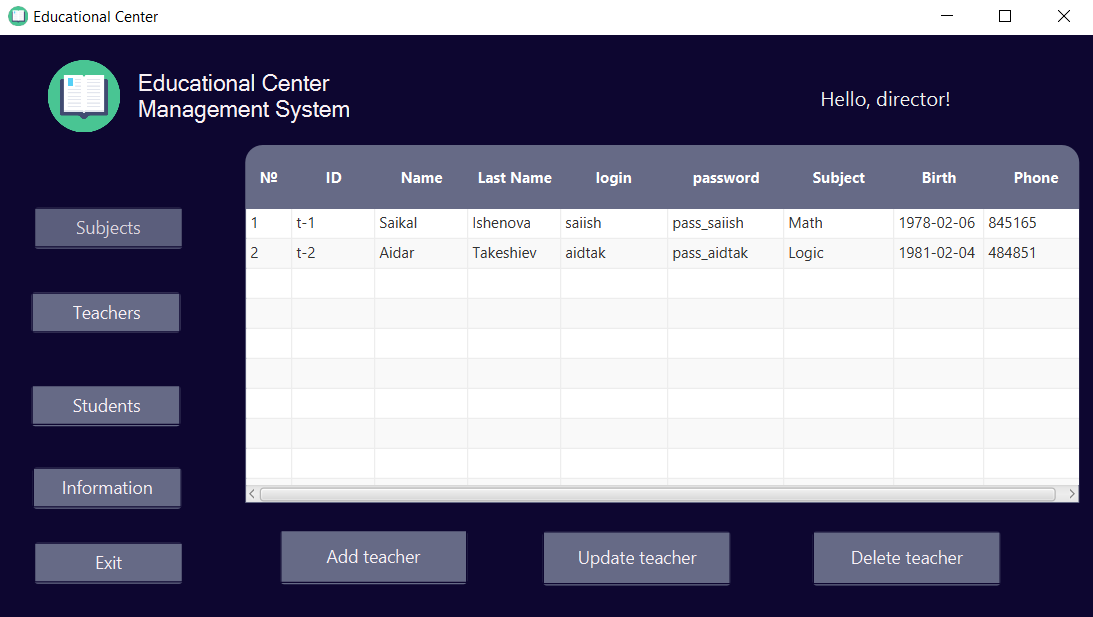
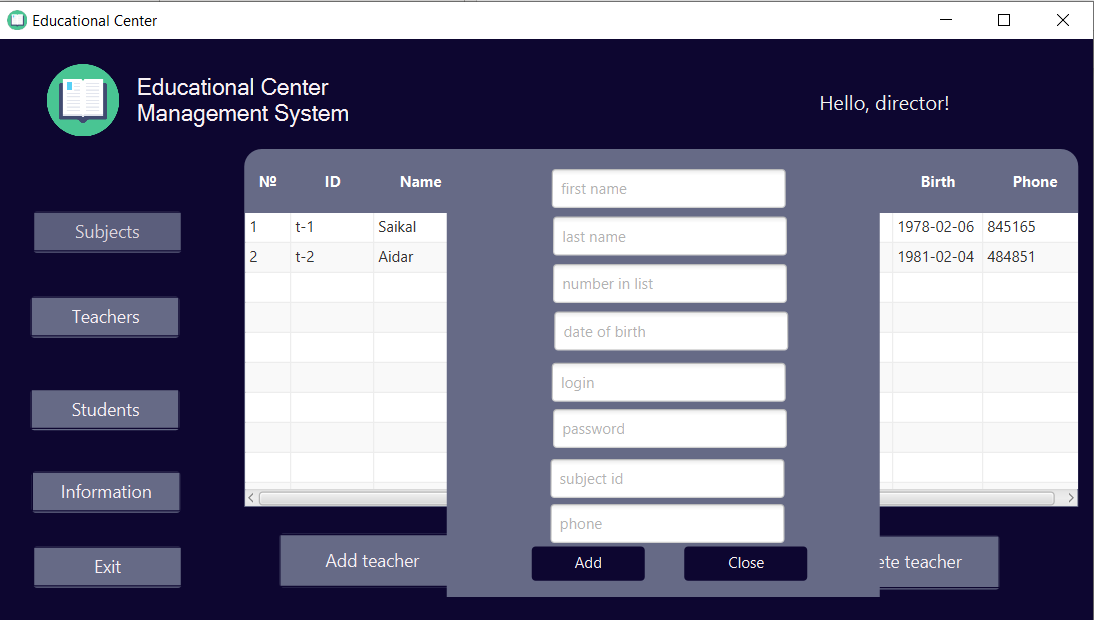
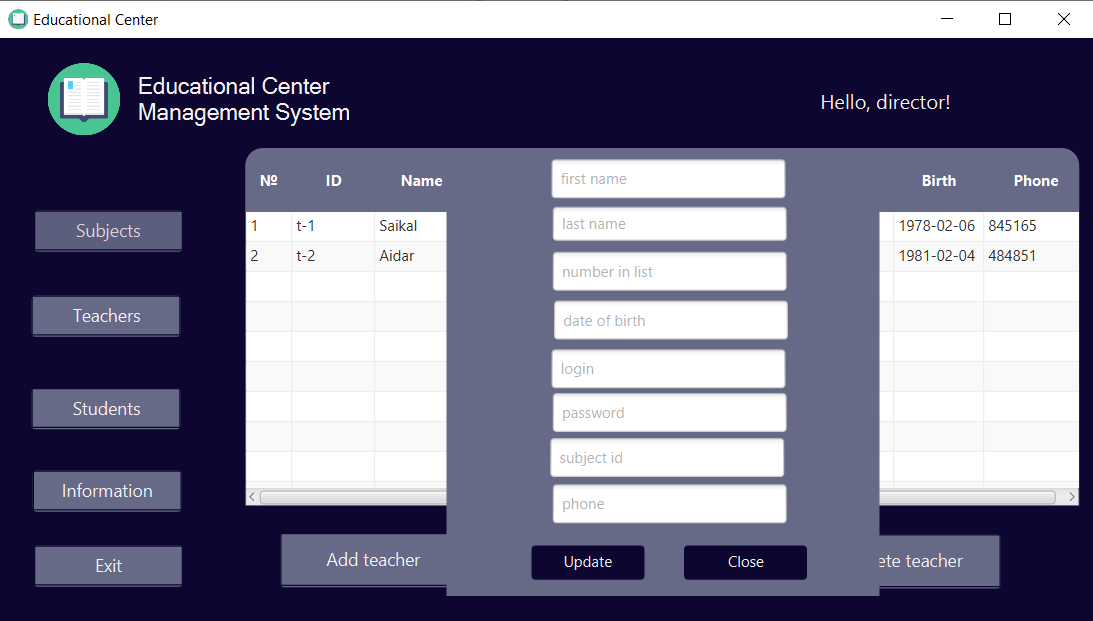




Subject page

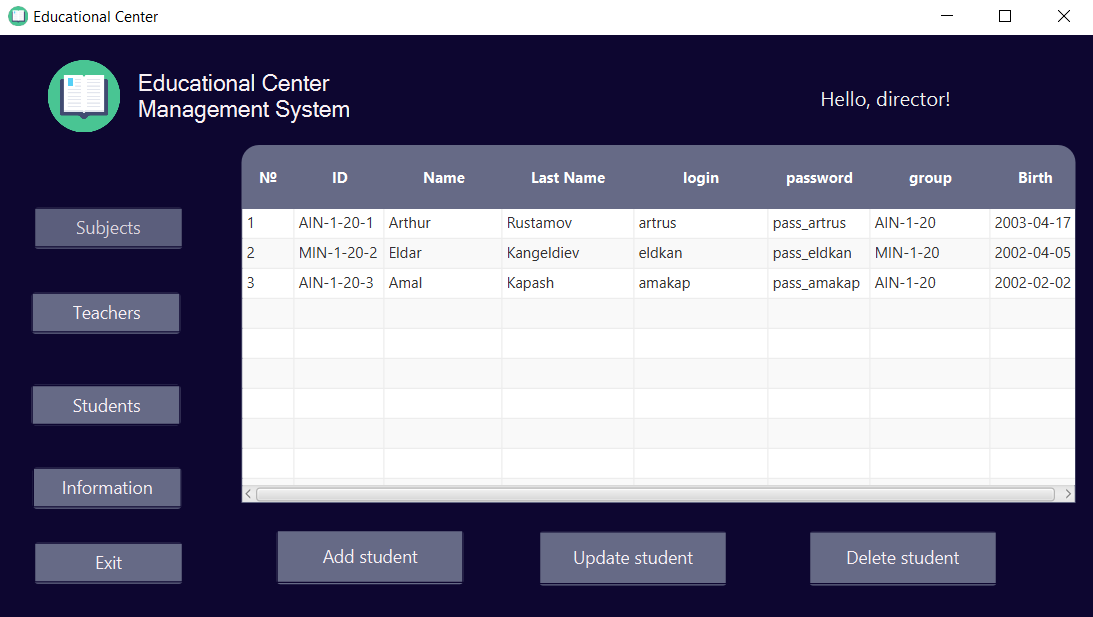
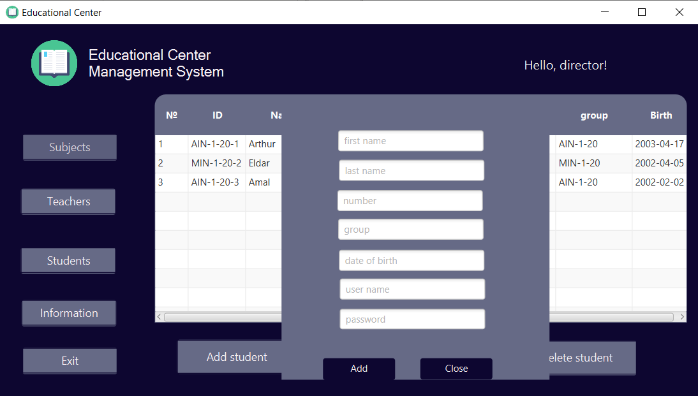
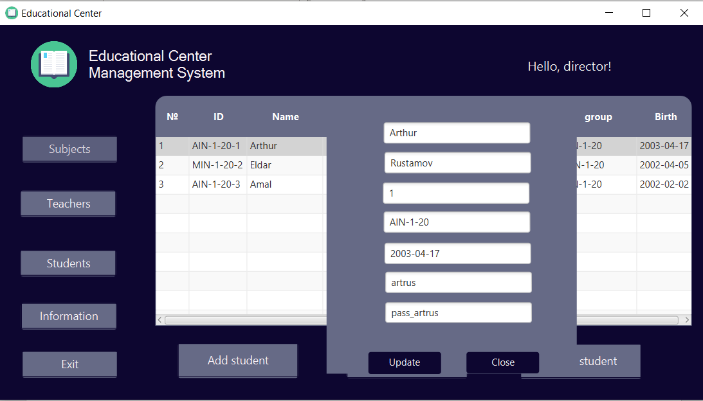
On the first page you can choose the subject and by the clicking on this button it will open you a table view with the list of students who are taking this classes. Also you can add a student to any subject by his ID.

Director’s account



Teachers’ list

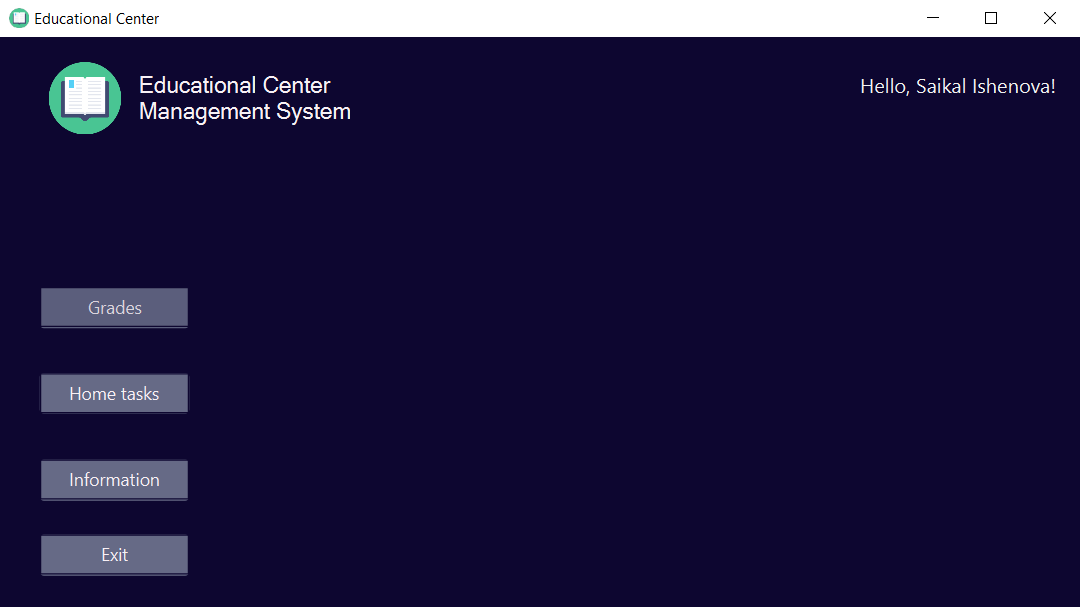
On this page you can see the list of all teachers. Here you can add a new teacher. Update or delete an information about selected one.



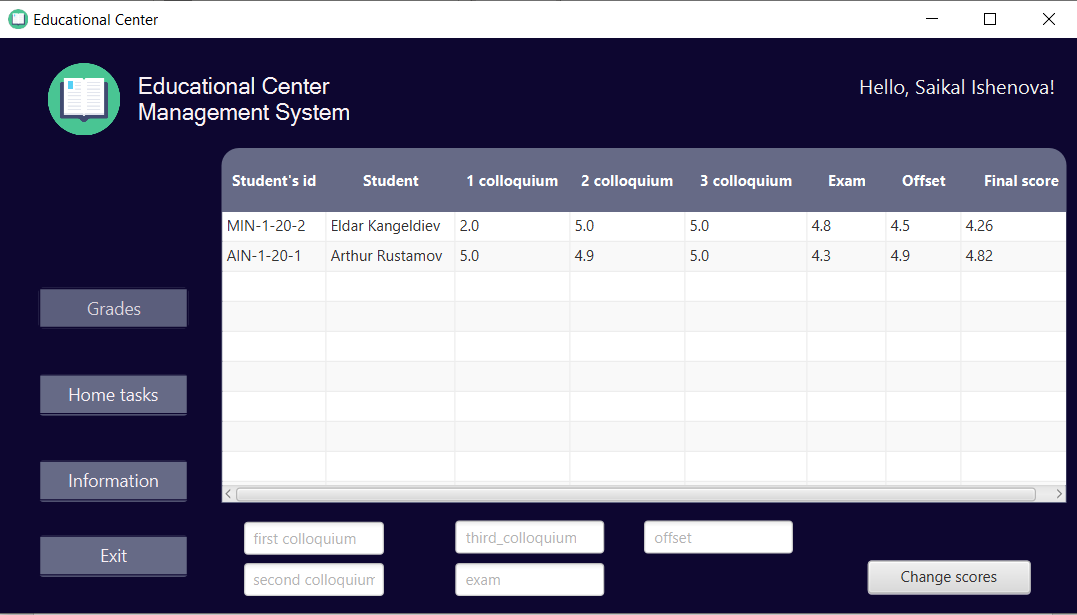
Students’ list

On this page you can see the list of all students. Here you can add a new teacher. Update or delete an information about selected one.

Teacher’s account

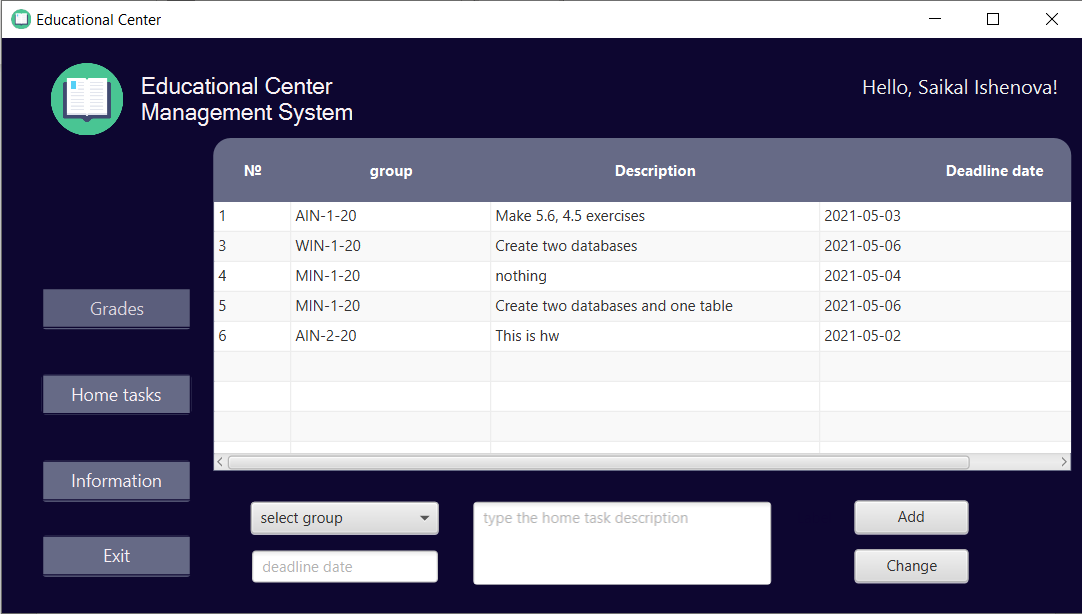


On this page you can choose the function you want, and it will show you another scene.



List of grades

On this page you can see the list of all grades of students you have.



Home works page

The list of all home works of your subject. Here you can add or update the information.

Student’s account

A screenshot of a computer

Description automatically generated with medium confidence

On this page you can choose the function you want, and it will show you another scene.

A screenshot of a computer

Description automatically generated with medium confidence

List of subjects

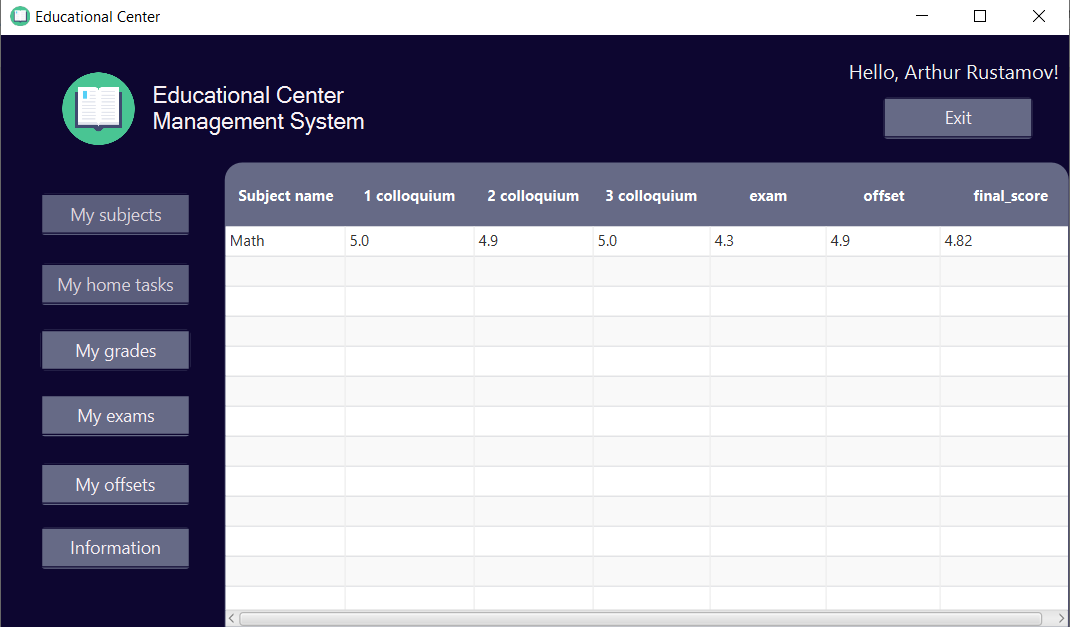
Here you can see all subjects you are taking now.

A screenshot of a computer

Description automatically generated with medium confidence

The list of home tasks

Here you can see the list of all home tasks you should do.



List of grades

Here you can see all the grades you have.

A screenshot of a computer

Description automatically generated

List of exams

Here you can see the list of your exams by the subject, the consultation date, the pass date and the score.

A screenshot of a computer

Description automatically generated

List of offsets

Here you can see the list of your offsets by the subject, the consultation date, the pass date and the score.

Vision

Business opportunity

The program has good perspectives for organisations chained with education. These companies can use this program for managing their stock, allowing their customers use the database and manage it for their needs. In addition, this program is able to show all information chained with user depending on a type of account.

Problem Statement:

|  |  |
| --- | --- |
| The problem of | The lack of knowledge  Slow work of the database  Working on different devices |
| Affects on | The speed of the work of the program  Updating the program  Productivity of the work |
| A successful solution would be | Studying and learning new libraries of the java (javafx, java.sql)  Working together to not to share all the time between different devices |

Risk Management

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Risk ID | Risk Description | Department | Consequence | Rank | Strategy |
| 1 | The code can be used incorrectly | All departments | Major | Medium Risk | Accept |
| 2 | There might be problems with authorization | All departments | Significant | Medium Risk | Control |
| 3 | Not every device can support this program | All departments | Major | Medium Risk | Accept |
| 4 | Information can be uncomfortably stored in database (mySQL) | All departments | Significant | Medium Risk | Control |

**Conclusion. Code Overview**

Text

Description automatically generated

We used different anchor panes to show the information we need or hide what we do not need.

Text

Description automatically generated

Using special SQL query we delete information we want from Database and update the table view.

Text

Description automatically generated

We get selected row from the table view using index, we created before.

Text

Description automatically generated

We get information and select an Update Query of SQL and “Where” clause to change specific row.

Text

Description automatically generated

We create an Insert Query to add a row to the table in mySQL Database.

Text

Description automatically generated

Using javafx.Initializable library we create @Override method to update table not using special buttons. We create a complicated Select Query with Inner Join to have all information from mySQL by foreign keys.

Text

Description automatically generated

These are most common libraries we used in our program.

Text

Description automatically generated

To fill the tables we created several java classes with getters, setters and constructors and used ObservableList to write this information.

Text

Description automatically generated

Such functions we used for searching the repeating cells as the Strings we wrote in our login and password TextFields. If there is only one repeated row, the method returns true.

Text

Description automatically generated

To change among different scenes, we wrote new method that open the scene by typing just the directory of the .fxml file.

Text

Description automatically generated

This method allows us to connect to mySQL by the name of the Database we need, our username and password. If everything is correct, it returns connection and prints “Connection Success”.